Realtime SMT Process RPM & Working Acknowledge

Tachometer

Tachometer

Laser and Detector

Laser and Detector

Raspberry Pi

Reflow Soldering Oven - 2

Stencil Separator

Automated Optical Inspection

Heater

Pre-Heater

Fluxer

Placement Machine -2

Solder Paste Inspection

Solder Paste Application

Visual Inspection

Reflow Soldering Oven

Placement Machine

Solder Paste Inspection

Solder Paste Application

PC Board Handlers

Monitoring using Absence of Solder Tube Output

Raspberry pi Cycle Time O/P & Count

Remote Desktop

& Arduino

**BLOCK DIAGRAM OF TACHOMETER**

RPM & LAMP ACKNOWLEDGE

RASPBERRY PI



**EXHAUST FAN**

**IR SENSOR**

**RPM CALCULATOR**

**ARDUINO UNO**

**LAMP**

Remote Desktop

**--**

**BLOCK DIAGRAM OF LASER DETECTOR**

**BLOCK DIAGRAM OF LASER DETECTOR**

RASPBERRY PI

LASER SOURCE

LASER DETECTOR

* A Pre-Determined height is set & the Laser-Detector is Setup.
* Initially, The Solder Tube Setup is full and blocks the Laser, so no Output is detected.
* Once the Solder Tube goes below the Set Level, Output is detected.
* An Indication is given to Refill the Solder Tube.

**BLOCK DIAGRAM OF COLOR SENSOR SETUP**

Raspberry Pi

Color Sensor (Exit)

Color Sensor (Start)

* Before the PCB enters the Solder Application, the Color sensor recognizes a particular color whose identity is already fed into the code this triggers the timer to START.
* The Average Cycle Time is found for that particular model and is preloaded onto the Raspberry Pi.
* Once the PCB exits after the AOI, the timing process for that particular PCB is clocked and compared with the Preloaded Time.
* If the Time exceeds the Preloaded Time then, that particular PCB is noted on a database. Simultaneous Monitoring and Timing is done for the forthcoming PCB’s.

**ABSTRACT :**

In PCB manufacturing process , there can be many obstacles to overcome because of the attention to detail involved and the natural complexity of the process makes it a delicate process . We have been provided such problems for a certain manufacturing unit in **VISTEON COMPANY** . The entire process is called ***SURFACE MOUNT TECHNOLOGY* (SMT) .**To understand the problems we need to know about the processes involved :

* **SOLDER PASTE PRINTING –** This process essentially uses a printer which uses a stencil & squeegees setup . The stencil has the blueprint to the circuit which will be imprinted on to the PCB .
* **COMPONENT PLACEMENT –** The required **SMD**’s decided upon design are carefully placed onto the PCB .
* **REFLOW SOLDERING –** The solder connections between the components and PCB are done here by heating in an oven-like setup. Exhaust fans are present to maintain the oven at the required temperature . This is a critical process .
* **AUTOMATED OPTICAL INSPECTION -** Necessary inspections are done and then the finished boards are sent over to the assembly unit **.**

**THE PROBLEMS STATED BY THE COMPANY WERE AS FOLLOWS :**

* In the **Solder paste printing** process , there is no indication on whether the solder tubes used by the process are present or not **.**
* In the **Reflow soldering** process , the exhaust fans mentioned above are crucial . In times of failure , there is no intimation whether the exhaust fans are running or not which will damage heating coils **.**
* Throughout the **SMT** process , data logging of unfinished products are being entered manually which may result in logging errors .

**We have found a viable solution for the problems stated above using Raspberry PI , Arduino and sensors which helps us provide live monitoring through IoT technology .**

**OBJECTIVE :**

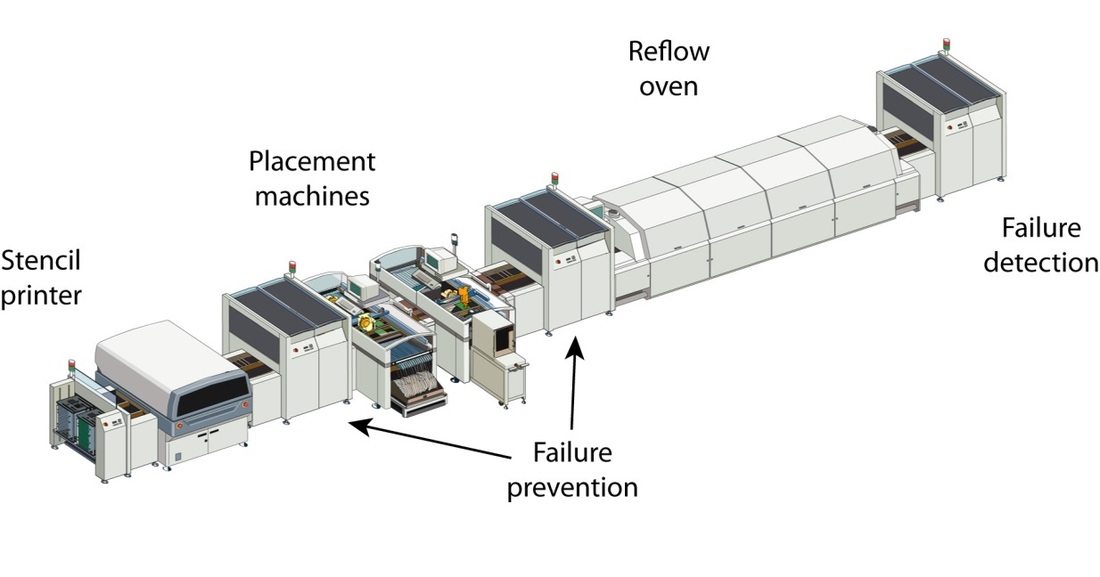
We shall discuss the solutions here to the above mentioned problems **.** We have tried to provide to solutions to both ours and company’s satisfaction**.**

* In the solder paste process, the indication used for detecting the presence of solder tube is done using a LASER – DETECTOR setup.
* For the reflow soldering process, we have done an indication setup using a TACHOMETER, which is powered by an ARDUINO CONTROLLER. The Tachometer setup is enabled by an IR Sensor which obtains input from the exhaust fan to calculate rpm and ensures whether it’s running or not , which is a non-contact type process .
* In the **SMT** process , two **bar code scanners** are placed one at the initial stage where the PCBs enter and another at the exit stage . This essentially records the **cycle time** and enables us to log the information related to the particular PCBs which are unfinished .
* So the above solutions are monitored by a **Raspberry Pi** and is communicated to the operator who is in charge of particular SMT line via IoT .

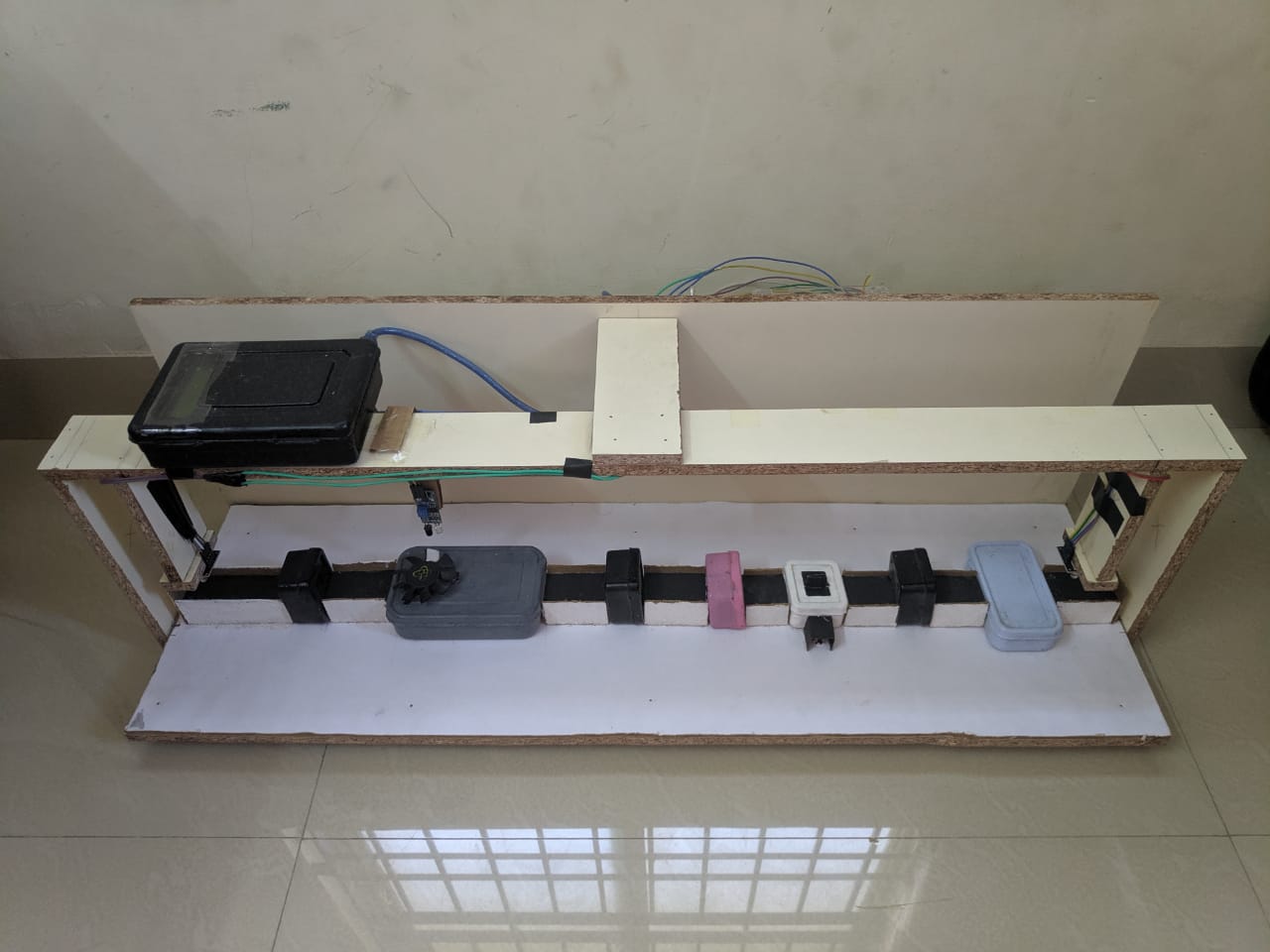
We shall discuss the solutions here to the above mentioned problems **.** We have tried to provide to solutions to both ours and company’s satisfaction **.**

* In the solder paste process , the indication used for detecting the presence of solder tube is done using a LASER – DETECTOR setup .
* For the reflow soldering process , we have done an indication setup using a TACHOMETER , which is powered by an ARDUINO CONTROLLER . The Tachometer setup is enabled by an IR Sensor which obtains input from the exhaust fan to calculate rpm and ensures whether it’s running or not , which is a non-contact type process .
* In the **SMT** process , two **colour sensors** are placed one at the initial stage where the PCBs enter and another at the exit stage . This essentially records the **cycle time** and enables us to log the information related to the particular PCBs which are unfinished.
* So the above solutions are monitored by a **Raspberry Pi** and is communicated to the operator who is in charge of particular SMT line via IoT.
* Since this is a prototype, I’ve done using cheaper viable options (sensors) and so we shouldn’t expect the same components to be used in real time application. My suggestions are to use a bar code scanner instead of the colour sensors, a highly accurate industrial tachometer and a computer much more powerful than the Raspberry Pi.

**SMT DIAGRAM:**

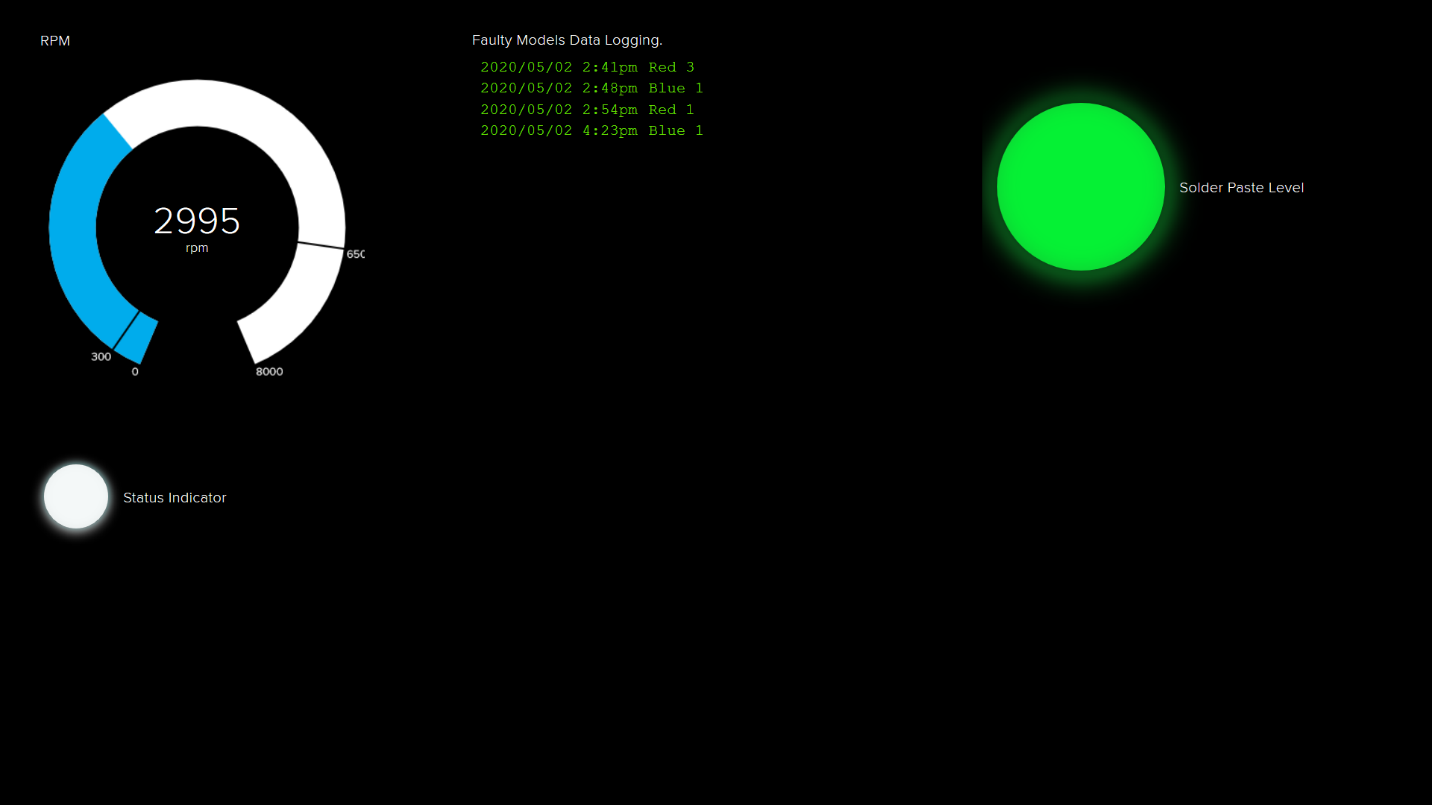


**Prototype:**



**IOT Interface :**

**Working state**

****

**Idle state**

****